



TITLE:

KETpicによる教材作成とSymbolic Thinking (数学ソフトウェアと教育 : 数学ソフトウェアの効果的利用に関する研究)

AUTHOR(S):

山下, 哲; 高遠, 節夫

CITATION:

山下, 哲 ...[et al]. KETpicによる教材作成とSymbolic Thinking (数学ソフトウェアと教育 : 数学ソフトウェアの効果的利用に関する研究). 数理解析研究所講究録 2012, 1780: 72-82

ISSUE DATE:

2012-03

URL:

<http://hdl.handle.net/2433/171831>

RIGHT:

K_ETpic による教材作成と Symbolic Thinking

木更津工業高等専門学校・基礎学系 山下 哲 (Satoshi Yamashita)

Fuculty of Fundamental Research,

Kisarazu National College of Technology

東邦大学・薬学部 高遠 節夫 (Setsuo Takato)

Fucluty of Pharmaceutical Science,

Toho University

1 はじめに

高専や大学初年級の数学教員は、授業で使用するオリジナル印刷教材を作成する際、数式を多用するため、Word や一太郎などのワープロよりも T_EX をよく利用する。数学の教材では、学生に新しい概念のイメージを把握させるために図を用いることが多いが、T_EX で作成した教材に図を挿入することは容易な作業ではない。この結果、図のない教材が多いという現状がある。このことは、2008 年に 56 高専・27 大学 667 名が回答した「授業での図の利用に関するアンケート」の結果からも明らかである ([12])。

そこで、T_EX 文書に正確で美しい図を容易に挿入するために、数式処理システム (Computer Algebra System, 略して CAS) のパッケージとして K_ETpic を 2006 年に開発した ([10])。開発当初は CAS として Maple を利用したが、その後、Mathematica, Maxima, Scilab, R でも利用できるようになった ([2], [3], [1], [9])。これらの CAS に対応した K_ETpic パッケージは、K_ETpic の Web ページ <http://ketpic.com/> より無料でダウンロード可能である。K_ETpic で作成された図の特長は、正確な長さの線画、豊富な 2D および 3D 描画表現、文字や数式が T_EX と同じ字体であるという親和性の高さなどが挙げられる (図 1, 図 2 参照)。最初の 2 つの特長は、CAS の数式処理機能を利用することにより実現できた。K_ETpic を利用して図を作成するには、以下の手順で作業すればよい (図 3 参照)。

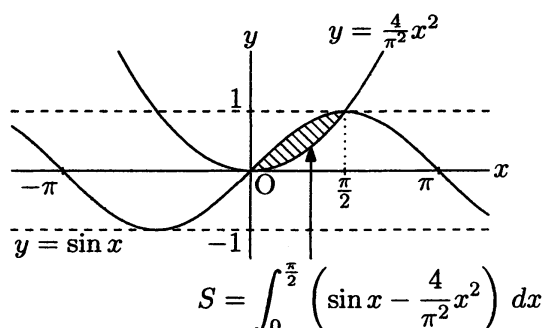


図 1. 2D 描画における斜線塗り

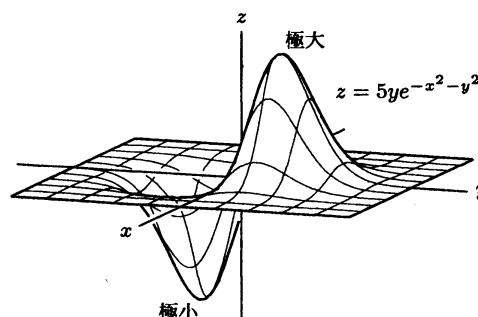


図 2. 2 変数関数の極大・極小

- (I) CAS を立ち上げ, $\text{K}\epsilon\text{T}\text{pic}$ を読み込み, 描画のプロットデータを作成する.
- (II) $\text{K}\epsilon\text{T}\text{pic}$ を用いて, プロットデータから $\text{L}\text{A}\text{T}\text{E}\text{X}$ 用描画コード (Tpic specials) を生成し, 図ファイルに書き出す.
- (III) TEX コマンド $\text{\textbackslash input}$ により, TEX 文書に図ファイルを挿入する.
- (IV) TEX をコンパイルし, プレビューである DVI (PDF) で確認する. 図の修正があれば, CAS に戻り, (I) から作業を繰り返す.

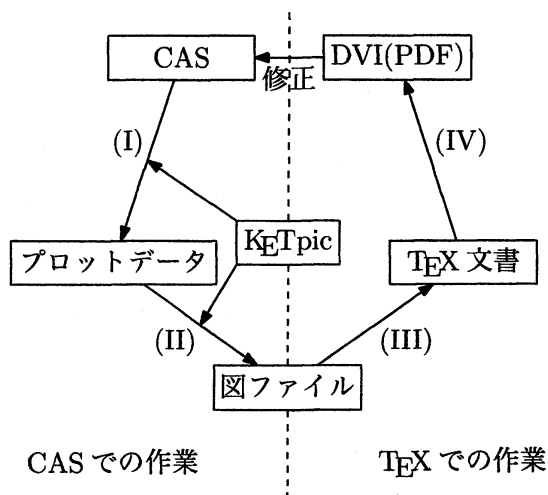


図 3. $\text{K}\epsilon\text{T}\text{pic}$ による教材作成の手順

この手順に従うと, 図を TEX 文書に簡単に挿入でき, 変更したい部分だけ図を修正すればよい. 以上のことから, $\text{K}\epsilon\text{T}\text{pic}$ は印刷教材に挿入する図の作成に適している ([11], [4], [5], [8], [6]).

一方, 図入り印刷教材を作成するためには, 描画機能だけでなく, 表作成機能・レイアウト機能・コマンド作成機能も必要である. 最近, $\text{K}\epsilon\text{T}\text{pic}$ にこれらの機能を新しく装備することができた. 本論文の第 2 節では, これらの機能に関してワープロと $\text{T}\text{E}\text{X}+\text{K}\epsilon\text{T}\text{pic}$ を比較し, どちらが図入り印刷教材の作成に適しているか検討する. 第 3 節では, Scilab 版 $\text{K}\epsilon\text{T}\text{pic}$ を用いて具体的な図入り印刷教材を作成しながら, $\text{K}\epsilon\text{T}\text{pic}$ に装備された新機能を紹介する. 第 4 節では, 第 3 節の教材作成事例を用いて, 図入り印刷教材の作成に必要な事項を明らかにし, その中でも Symbolic Thinking に焦点を当てる. 第 5 節では, 本論文の内容をまとめ, $\text{K}\epsilon\text{T}\text{pic}$ に関する今後の課題について述べる.

2 ワープロと $\text{T}\text{E}\text{X}+\text{K}\epsilon\text{T}\text{pic}$ の比較—教材作成の視点から—

数学の図入り印刷教材を作成するためには, 文書を作成するエディタと図を作成する描画用ソフトが必要である. 主なエディタとしては, Word や一太郎などのワープロ, または組版ソフト TEX がよく利用されている. また, 主な描画用ソフトとしては, 花子や Illustrator などのお絵描きソフト, または Gnuplot や CAS などのグラフ作成ソフトがよく利用されている. TEX との親和性の高さから, WinTpic や初等数学プリント作成マクロ emath の描画用スタイルファイル emathP を利用する人も少なくない. $\text{K}\epsilon\text{T}\text{pic}$ は, エディタとして TEX , 描画用ソフトとして CAS の利用を前提に開発されたツールである. 本節では, ワープロ Word と組版ソフト TEX という 2 種類のエディタに着目して, どちらが図入り印刷教材の作成に適しているか検討する.

図入り印刷教材を作成するために必要となる主な機能は, 数式表示機能, 描画機能, 記号作成機能, 表作成機能, レイアウト機能である. これらの機能について, Word と TEX を比較すると表 1 のようになる. Word は, 数式表示機能・描画機能・表作成機能

表 1. 図入り教材作成機能に関する Word と $\text{T}_\text{E}\text{X}$ の比較

必要な機能	Word	$\text{T}_\text{E}\text{X}$	手書き
数式表示	△	○	○
描画	△	△	△
記号作成	○	×	○
表作成	△	△	○
レイアウト	○	△	○

○：使いやすい，△：やや使いにくい，×：使いにくい

でやや使いにくい点があるが，全体的には図入り教材の作成に適したエディタであるといえる．一方， $\text{T}_\text{E}\text{X}$ は，記号作成機能で $\text{T}_\text{E}\text{X}$ マクロを作成しなければならないため，初心者にとって大変難しい．また， $\text{T}_\text{E}\text{X}$ の表作成機能・レイアウト機能では， $\text{T}_\text{E}\text{X}$ が図表や文字・数式の大きさを自動計算して，図の配置，罫線の位置，字間・行間などを自動で調整してしまう．この自動調整されたレイアウトは論文作成に適しているが，学生にわかりやすい教材のレイアウトとしては的確ではない．また，教材作成者が思い通りのレイアウトに修正することが難しい．この点で， $\text{T}_\text{E}\text{X}$ はこれらの機能についてやや使いにくく，全体的には図入り教材の作成に適していない．

以上のことから，図入り印刷教材作成の適性を比較すると，Word は $\text{T}_\text{E}\text{X}$ よりも総合的に優れていることがわかる．ちなみに，表 1 では手書きも比較してみた．手書きは，描画機能以外のどの機能についても思い通りにできる．ただし，正確な図を描くことが難しいため，描画機能についてはやや劣る．手書きは旧式の方法であり，コンピュータが普及した現在ではほとんど利用されないが，図入り印刷教材の作成に関して非常に優れていることがわかる．

では，Word と $\text{T}_\text{E}\text{X}+\text{K}_\text{E}\text{Tpic}$ で図入り印刷教材作成の適性を比較してみよう．すると，表 2 のようになる．描画機能については， $\text{K}_\text{E}\text{Tpic}$ を用いることにより，図 1 や図 2 のような高品質なレベルまで改善される．2010 年以降， $\text{K}_\text{E}\text{Tpic}$ に $\text{T}_\text{E}\text{X}$ コマンド作成機能，

表 2. 図入り教材作成機能に関する Word と $\text{T}_\text{E}\text{X}+\text{K}_\text{E}\text{Tpic}$ の比較

必要な機能	Word	$\text{T}_\text{E}\text{X}+\text{K}_\text{E}\text{Tpic}$	手書き
数式表示	△	○	○
描画	△	○	△
記号作成	○	○	○
表作成	△	○	○
レイアウト	○	○	○

○：使いやすい，△：やや使いにくい，×：使いにくい

表作成機能、レイアウト機能が追加され、 $\text{T}_{\text{E}}\text{X}$ 単体では不十分であった機能についてかなり改善された。以上のことから、図入り印刷教材作成の適性について、 $\text{T}_{\text{E}}\text{X}+\text{K}_{\text{E}}\text{Tpic}$ は Word よりも優れており、手書きも凌駕している。 $\text{K}_{\text{E}}\text{Tpic}$ は $\text{T}_{\text{E}}\text{X}$ 総合支援ツールとして進化したといえる。

3 $\text{K}_{\text{E}}\text{Tpic}$ の新機能について

本節では、2010 年以降に開発された $\text{K}_{\text{E}}\text{Tpic}$ の新機能について、Scilab 版 $\text{K}_{\text{E}}\text{Tpic}$ による図入り印刷教材の具体的な作成例を通して紹介する。本節で使用する環境は以下の通りである。

OS : Windows (XP, Vista, 7)
 CAS : Scilab5.3.3
 $\text{T}_{\text{E}}\text{X}$: $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$
 $\text{T}_{\text{E}}\text{X}$ エディタ : EasyTeX
 $\text{K}_{\text{E}}\text{Tpic}$: ketpicsciL5
 $\text{T}_{\text{E}}\text{X}$ スタイルファイル : ketpic, ketlayer, emath, emathMw

3.1 図入り印刷教材用 $\text{T}_{\text{E}}\text{X}$ 文書の作成

本節では、内分点の位置ベクトルの公式に関する図入り印刷教材を作成する（図 4 参照）。この公式について、教科書を用いて説明すると、線分 AB の比 $m:n$ から分子を $\overrightarrow{mOA} + \overrightarrow{nOB}$ と間違ってしまう学生が 35% 程度いる。そこで、本教材の目的は、 $m:n = 3:2$ という具体的な場合で教材の図の破線を利用して、分子の正確な形 $\overrightarrow{nOA} + \overrightarrow{mOB}$ を学生にイメージさせることである。本教材を用いることにより、前述の間違いをする学生が 5% 程度まで減少した。

本教材の $\text{T}_{\text{E}}\text{X}$ 文書 material.tex を作成してみると、次のようになる。

```

1 \documentclass[a4j]{jarticle}
2 \usepackage{ketpic, ketlayer}
3 \usepackage{emath, emathMw}
4 \def\sikaku{\input{sikaku.tex}}
5 \begin{document}
6 \begin{layer}{120}{80}
7 \putnotesw{115}{10}{\input{figure.tex}}
8 \putnotes{95}{60}{\bf \$\bm{m=3}$, \$\bm{n=2}$の場合}
9 \end{layer}
10 \begin{mawarikomi}[9]{55mm}{}
11 2点 A, B に対し、線分 AB を $m:n$ の比に内分する点 P の位置ベクトル
    \$\bekutoru{OP}$は

```

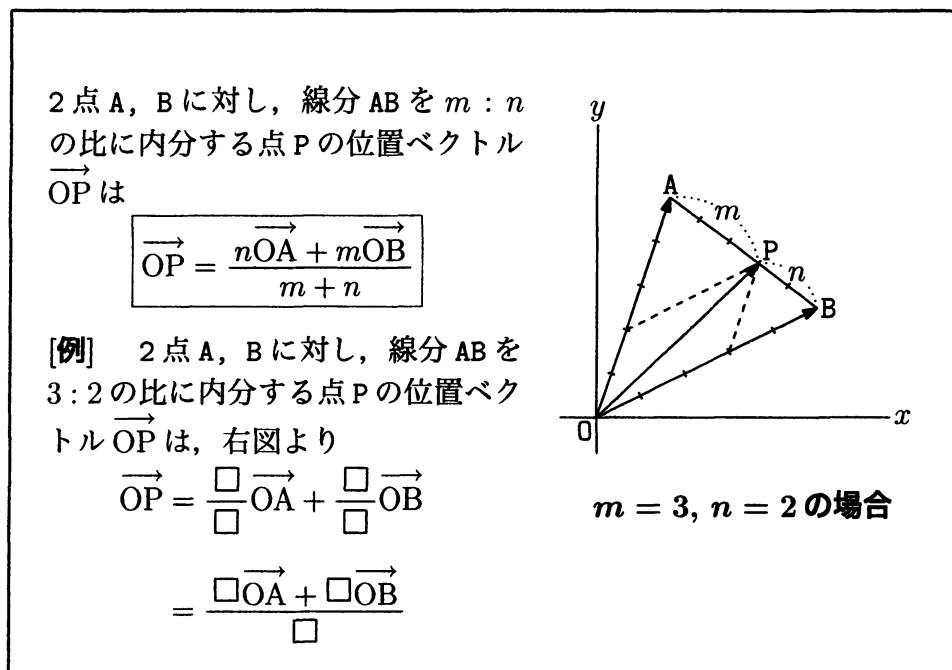


図 4. 内分点の位置ベクトルの公式に関する図入り教材

```

12 \hspace{3zh}\fbox{
    $\displaystyle \overrightarrow{OP} = \frac{n\overrightarrow{OA} + m\overrightarrow{OB}}{m+n}$
  } \vspace{3mm}
13 {\bf [例]} 2点 A, B に対し, 線分 AB を $3:2$ の比に内分する
    点 P の位置ベクトル $\overrightarrow{OP}$ は, 右図より
14 \hspace{10mm} $\displaystyle \overrightarrow{OP} = $
    \bunsuu{\syikaku}{\syikaku}\overrightarrow{OA} +
    \bunsuu{\syikaku}{\syikaku}\overrightarrow{OB}$ \vspace{3mm}
15 \hspace{10mm} $\displaystyle \phantom{\overrightarrow{OP}} = $
    \bunsuu{\syikaku\overrightarrow{OA} + \syikaku\overrightarrow{OB}}{\syikaku}$
16 \end{mawarikomi}
17 \end{document}

```

1 行目は $\text{T}_{\text{E}}\text{X}$ 文書の書式を定義し, 2 行目と 3 行目で使用する $\text{T}_{\text{E}}\text{X}$ スタイルファイルを読み込む. 4 行目でオリジナルの $\text{T}_{\text{E}}\text{X}$ コマンド `\syikaku` を定義し, 絵文字 \square を表示させることができる. このコマンドの作成については, 次節で説明する. ここまでを**プリアンブル部**といい, $\text{T}_{\text{E}}\text{X}$ 文書の書式の定義 ($\text{T}_{\text{E}}\text{X}$ マクロ) を記述する部分である. 5 行目で $\text{T}_{\text{E}}\text{X}$ 文書の本文を書き始め, 13 行目で本文を終了する.

本文の記述は以下の通りである. 6 行目で $\text{T}_{\text{E}}\text{X}$ スタイルファイル `ketlayer.sty` の `layer` 環境を始め, 7 行目で図挿入し, 8 行目でキャプションを挿入し, 9 行目で `layer` 環境を終了する. `layer` 環境の使い方については, 第 3.4 節で詳しく述べる.

10 行目で $\text{T}_{\text{E}}\text{X}$ スタイルファイル `emathMw.sty` の `mawarikomi` 環境を始める. この環境により, 文章の右側に指定した幅の空白部分を作ることができる. 6 行目の第 1 引数 [9] と第 2 引数 {55mm} は, 左側 9 行分の文章の右側に 55mm の幅でスペースを空ける.

第3引数 {} には、本来、文章の右側に入れる図ファイルを指定するのであるが、layer環境を用いて図を入れるため、このまま空欄で使用する。16行目で mawarikomi 環境を終了する。

11行目から15行目までに現れる \yenbekutoru と \yenbunsuu は emath コマンドであり、図4の教材の文章にあるようなベクトルと分数の記号で表示される。また、14行目と15行目には、オリジナルの T_EX コマンド $\yen sikaku$ を利用している。コンパイルして表示させると、図4の表示になる。

3.2 オリジナル T_EX コマンドの作成

絵文字□の図ファイル sikaku.tex を Scilab で以下のように作成する。

```
1 Ketlib=lib('c:/work/ketsciL5/');
2 Ketinit();
3 Setwindow([-0.5,3.5],[-0.5,3.5]);
4 G1=Framedata([1.5,1.5],1.5);
5 Openfile('sikaku.tex');
6 Beginpicture('1mm');
7 Drwline(G1);
8 Endpicture(0);
9 Closefile();
```

1行目で K_ETpic を読み込み、2行目で K_ETpic を初期化する。3行目で表示する範囲を指定する。第1引数のベクトル [-0.5,3.5] は x 座標の範囲 $-0.5 \leq x \leq 3.5$ を表し、第2引数のベクトル [-0.5,3.5] は y 座標の範囲 $-0.5 \leq y \leq 3.5$ を表す。第4行目で1辺の長さ3の正方形を定義する。第1引数のベクトル [1.5,1.5] は正方形の中心の座標を表し、第2引数 1.5 は x 軸および y 軸方向に ± 1.5 だけ中心から離れた正方形を描くことを表す。5行目で図ファイル sikaku.tex を開き、6行目以降を書き出す。6行目で単位長を 1mm に指定して picture 環境を始める。7行目で1辺の長さ 3mm の正方形を描き、8行目で picture 環境を終了する。8行目の引数 0 は、座標軸を描かずに終了することを意味する。9行目で図ファイル sikaku.tex を閉じる。

3.3 教材の図の作成

図4の教材の図ファイル figure.tex を Scilab で以下のように作成する。

```
1 Ketlib=lib('c:/work/ketsciL5/');
2 Ketinit();
3 Setwindow([-0.5,4],[-0.5,4]);
4 P0=[0,0];PA=[1,3];PB=[3,1.5];
5 G1=Listplot([PA,PB]);
6 NbL1=Naibun(PA,PB,5);NbpL1=NbL1(1);TmL1=NbL1(2);
7 P11=NbpL1(1); P12=NbpL1(2); P13=NbpL1(3); P14=NbpL1(4);
```

```

8 B1=Bowdata(P13,PA,2,0.5); B2=Bowdata(PB,P13,2,0.5);
9 NbL2=Naibun(P0,PA,5);NbpL2=NbL2(1);TmL2=NbL2(2);
10 P22=NbpL2(2);
11 NbL3=Naibun(P0,PB,5);NbpL3=NbL3(1);TmL3=NbL3(2);
12 P33=NbpL3(3);
13 D1=Listplot([P22,P13]); D2=Listplot([P33,P13]);
14 Openfile('figure.tex');
15 Beginpicture('1cm');
16 Drwline(G1,TmL1,TmL2,TmL3);Dottedline(B1,B2);Dashline(D1,D2);
17 Bowname(B1,'m');Bowname(B2,'n');
    Letter(PA,'n','A'); Letter(PB,'e','B'); Letter(P13,'ne','P');
18 Arrowline(P0,PA);Arrowline(P0,PB);Arrowline(P0,P13);
19 Endpicture(1);
20 Closefile();

```

3行目から13行目で図のプロットデータを作成し、14行目から20行目で作成したプロットデータを図ファイル figure.tex に書き出す。4行目で3点 O, A, B の座標を指定し、5行目で線分 AB のプロットデータを作成する。6行目でオリジナルの関数 Naibun を用いて、線分 AB 上に付ける5等分目印のプロットデータを作成する（9行目と11行目も同様である）。ここで、NbpL1 は線分 AB の5等分点の座標のリストで、TmL1 は5等分目印のプロットデータのリストである。8行目で、内分点 P13 から点 A と点 B までの弓形のプロットデータを作成する。13行目で、本教材の最重要ポイントとなる破線のプロットデータを作成する。

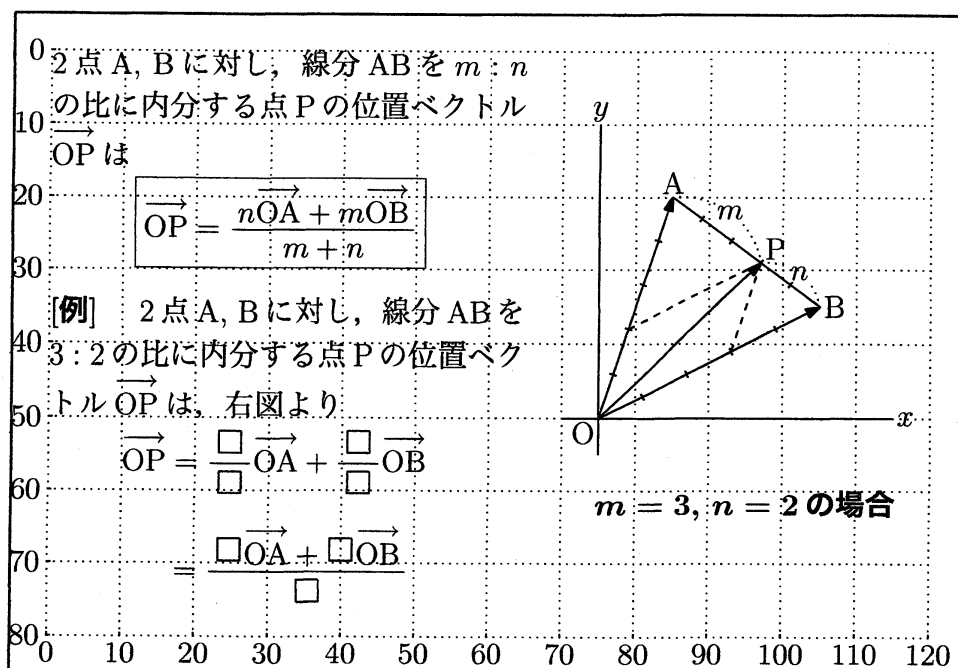
Scilab の関数 function を用いて、オリジナル関数 Naibun を以下のように定義する。

```

1 function Out=Naibun(PA,PB,N)
2 NbpL=list();TmL=list();
3 L=0.1;
4 Vab=PB-PA;Vn=[-Vab(2),Vab(1)];Un=Vn/norm(Vn);
5 for I=1:N-1
6 Pi=(N-I)/N*PA+I/N*PB;NbpL($+1)=Pi;
7 P1=Pi+Un*L/2;P2=Pi-Un*L/2;Ti=Listplot([P1,P2]);TmL($+1)=Ti;
8 end
9 Out=list(NbpL,TmL);
10 endfunction

```

1行目で、3つの引数をもつ関数 Naibun を定義し、Out で出力することを宣言する。第1引数と第2引数は線分の両端点 A, B の座標で、第3引数 N は N 等分を表す。2行目で等分点の座標のリスト NbpL と等分目印のプロットデータのリスト TmL を空リストにす

図 5. K_ETpic の layer 環境

る. 3 行目で等分目印の長さ L の値を設定する. 4 行目の V_{ab} は \vec{AB} であり, U_n は \vec{AB} に垂直な単位ベクトルになる. 5 行目から 8 行目の for 文で, N_{bpL} に等分点の座標を, T_{mL} に等分目印のプロットデータを追加していく. 9 行目で最終の N_{bpL} と T_{mL} のリストを Out として出力する.

3.4 K_ETpic のレイアウト機能について

最後に, 作成した図ファイル figure.tex を適切に配置するために, K_ETpic のレイアウト機能である layer 環境を利用する. p.4 の T_EX 文書 material.tex の 2 行目でスタイルファイル ketlayer を読み込んでいる. これで, layer 環境が使えるようになる. 6 行目で layer 環境を始め, 図 4 の教材上に幅 120 匁, 高さ 80 匁の罫線が表示される. 7 行目で表示された罫線上の座標 [115,10] の南西方向に figure.tex の図を配置する. 8 行目で罫線上の座標 [95,60] の南方向にキャプション「 $m=3, n=2$ の場合」を配置する. 9 行目で layer 環境を終了する. コンパイルして表示すると, 図 5 のようになる. 表示した罫線を除去するには, 罫線の高さを表す 1 行目の第 2 引数 {90} を {0} に変更すればよい.

3.5 表の作成

T_EX の array 環境を用いると, 表 2 の左表のようになる. 各セルの数字や記号の大きさを計測し, セルの幅や高さを自動調整するように設定されてしまい, 罫線が等間隔でない不格好な表となっている. そこで, K_ETpic のコマンド Tabledata を用いると, 表 2 の右表のような等間隔の表が思い通りに作成できる. この表ファイル zougen.tbl.tex

表 3. 関数の増減表

x	\cdots	$-\frac{1}{2}$	\cdots	0	\cdots
y'	$+$	0	$-$	0	$+$
y	\nearrow	$\frac{9}{8}$	\searrow	0	\nearrow

x	\cdots	$-\frac{1}{2}$	\cdots	1	\cdots
y'	$+$	0	$-$	0	$+$
y	\nearrow	$\frac{9}{8}$	\searrow	0	\nearrow

を Scilab で以下のように作成する.

```

1 Row1=list(10,10,10,10,10,10);
2 Col1=list(10,10,10);
3 Tb1=Tabledata([-1,1],Row1,Col1);
4 Openfile('zougen_tb1.tex');
5 Beginpicture('1mm');
6 Drwline(Tb1(1));
7 Putrow(Tb1,1,'c','$x$','$-\frac{1}{2}$','$\cdots$','$1$',
'$\cdots$');
8 Putrow(Tb1,2,'c','$+$','$0$','$-$','$0$','$+$');
9 Putrow(Tb1,3,'c','$\nearrow$','$\frac{9}{8}$','$\searrow$','$0$',
'$\nearrow$');
10 Endpicture(0);
11 Closefile();

```

1 行目で 6 列の幅を 10 区ずつに指定し, 2 行目で 3 行の高さを 10 区ずつに指定する. 3 行目で表の罫線のリスト Tb1 を作成する. 第 1 引数のベクトル [-1,-1] は表の表示する範囲を表の大きさに合わせることを意味する. 6 行目で表の罫線のプロットデータ Tb1(1) を実線で引く. 7 行目で表の 1 行目の各セルに左から順に書き出す. 第 2 引数 1 で表の行数を指定し, 第 3 引数 'c' で各セルに中央揃えで書き出すことを意味する.

4 図入り印刷教材の作成に必要な事項について

数学の図入り印刷教材の作成には, 作成者の数学的知識と数学教育実践や, 短時間で容易に作成できるツールが当然必要である. これら以外にどのような必要事項があるか考察してみよう. 紙と鉛筆を用いて手書きで作図する場合, 線を引く, マークを付けるなど, 一つ一つの作業を繰り返しながら, どのようにすれば的確な図が描けるか思考し, 作図を完成させていく. 作図ツールでは, 作図の各作業を記号化 (コマンド化) し, 記号の生成や記号による作業内容の記述を繰り返し, 作図を完成させる. 作成者は, 記号による一連の図作成作業の中で, 推論しながら, 新しい記号の生成, 記号の統合, 記号の再構成が繰り返し行われる. このような思考を symbolic thinking といい, 手書きの作図における思考と同様である. 最近, 作図ツールを利用する際, symbolic thinking を中断させることなく作業できることが必要であることがわかってきた. つまり, 手書き感覚で作図ツールが利用できることが必要がある. K_ETpic による作図はこのことを実

現している。本節では、第3.3節 p.6で作成した教材の図ファイル figure.tex をもとにして、symbolic thinking がどのように行われているか調査する。

3行目で図の表示範囲を指定し、4行目で3点 O, A, B の座標を適切な位置に決め、5行目で線分 AB のプロットデータを作成する。次に、線分 AB を5等分した等分点の座標と等分目印が必要になる。そこで、p.7の関数 Naibun を定義する必要がある。これは、symbolic thinking によって、新しい記号の生成が要請されることを意味する。関数 Naibun の定義の3行目で等分目印の長さを変数 L で定義しておき、修正する必要がある。容易に長さを変更できるようにしている。これは、修正の際、symbolic thinking を中断させないための工夫である。等分目印のプロットデータを作成するには、4行目で等分目印の単位方向ベクトルを求め、7行目で等分点の座標 P_i から $\pm \frac{L}{2}$ 倍して等分目印の端点 P_1, P_2 を求めている。ここでは、数学的知識に基づいて symbolic thinking を行いながら、記号の生成や記号による作業内容の記述が行われている。p.6 の figure.tex の作成に戻り、6行目で関数 Naibun を利用して、等分点の座標のリスト NbpL1 と等分目印のプロットデータのリスト TmL1 を作成する。同様にして、以下の行では、作図に必要なプロットデータが次々に作成されていく。14行目以降の図ファイル figure.tex への書き出しの作業では、手書きで図を描くように、作成したプロットデータにより線を書き出し、的確な位置に文字を配置していく。

以上のことから、第3.3節のプログラムは symbolic thinking を中断させないように配慮して設計されている。これが、 $\text{K}_\text{E}\text{T}_{\text{pic}}$ を手書き感覚で利用できる作図ツールになっている。

5 まとめと今後の課題

以上のことから、エディタとして $\text{T}_\text{E}\text{X}$ 、描画用ソフトとして CAS を利用し、さらに新機能を装備した $\text{K}_\text{E}\text{T}_{\text{pic}}$ を用いると、数学の図入り印刷教材が容易に作成できるようになった。作成した教材の図から学生が新しい概念をイメージし、知識として定着するためには、教材の図は概念を的確に表現するだけでなく、インパクトのあるものでなければならない。このことに集中して図を作成するためには、symbolic thinking を中断させることなく行える作成環境が必要となる。 $\text{K}_\text{E}\text{T}_{\text{pic}}$ はそれを可能とするツールとなっている。

今後の課題は、documentational approach の手法を用いて、図入り教材の作成に必要な能力を綿密に調査し、 $\text{K}_\text{E}\text{T}_{\text{pic}}$ を図入り教材作成ツールとして開発していくことである。また、授業デザインに基づいた図入り教材を $\text{K}_\text{E}\text{T}_{\text{pic}}$ で作成し、授業での実践による改良を行いながら、今までにない図入り教材に仕上げていくことである。

参考文献

- [1] Abe T., Fukazawa K., Kaneko M., Koshikawa H., Yamashita S. & Takato S.: "Migration of $\text{K}_\text{E}\text{T}_{\text{pic}}$ to Scilab and Comparison of Scilab with other CASs",

- 日本数学教育学会高専・大学部会論文誌, 第16巻, 第1号, pp.97–106, 2009.
- [2] 深澤謙次, 阿部孝之, 金子真隆, 関口昌由, 田所勇樹, 山下哲, 高遠節夫:「 $\text{K}\epsilon\text{T}\pi\text{c}$ の Mathematica への移植」, 木更津工業高等専門学校紀要, 第41号, pp.75–84, 2008.
 - [3] 深澤謙次, 阿部孝之, 金子真隆, 北原清志, 山下哲, 高遠節夫:「 $\text{K}\epsilon\text{T}\pi\text{c}$ の Maxima への移植と SAGE への移植の試み」, 京都大学数理解析研究所講究録, 1647, 数式処理と教育, pp.1–10, 2010.
 - [4] 金子真隆, 阿部孝之, 関口昌由, 山下哲, 高遠節夫:「 $\text{K}\epsilon\text{T}\pi\text{c}$ による曲面描画と教育利用」, 京都大学数理解析研究所講究録 1624, pp.1–10, 2009
 - [5] 金子真隆, 阿部孝之, 山下哲, 泉源, 深澤謙次, 北原清志, 高遠節夫:「線形代数の教科書における挿図の利用について— $\text{K}\epsilon\text{T}\pi\text{c}$ 利用の可能性を中心に—」, 京都大学数理解析研究所講究録 1674, pp.12–25, 2010
 - [6] Kaneko M., Abe T., Sekiguchi M., Tadokoro Y., Fukazawa K., Yamashita S. & Takato S.: “CAS-aided Visualization in $\text{L}\text{T}\text{E}\text{X}$ documents for Mathematical Education”, Teaching Mathematics and Computer Science (ISSN 1589-7389), Vol. VIII, Issue I, pp.1–18, 2010
 - [7] 金子真隆, 北原清志, 高遠節夫, 深澤謙次, 山下哲 (CASTeX 応用研究会編):「 $\text{K}\epsilon\text{T}\pi\text{c}$ で楽々 TEX グラフ」, イーテキスト研究所, 2011.
 - [8] 北原清志, 阿部孝之, 金子真隆, 山下哲, 高遠節夫:「全微分における図入り教材の作成例とその研究授業報告」, 京都大学数理解析研究所講究録 1674, pp.132–145, 2010
 - [9] Ouchi S. & Takato S.: “High-quality statistical plots in $\text{L}\text{T}\text{E}\text{X}$ for mathematics education using R-based $\text{K}\epsilon\text{T}\pi\text{c}$ plug-in”, Proc. 15th ATCM, pp.266–275, 2010.
 - [10] Sekiguchi M., Yamashita S. & Takato S.: “Development of a Maple Macro Package Suitable for Drawing Fine TEX -Pictures”, ICMS 2006, LNCS 4151, pp.24–34, Springer-Verlag, 2006.
 - [11] 高遠節夫, 阿部孝之, 泉源, 金子真隆, 北原清志, 関口昌由, 深澤謙次, 山下哲:「授業効果を高める挿図教材の作成」, 日本数学教育学会高専・大学部会論文誌, Vol.15, No1, pp.109–118, 2008.
 - [12] 高遠節夫, 山下哲, 金子真隆, 北原清志:「授業での図の利用に関するアンケート調査について」, 東邦大学教養紀要, 第42号, pp.31–40, 2011.